

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-91079

(43) 公開日 平成9年(1997)4月4日

(51) Int.Cl.⁸

識別記号 庁内整理番号
330

F I
G 0 6 F 3/03

330B

技術表示箇所

審査請求 未請求 請求項の数4 OL (全 25 頁)

(21)出願番号 特願平7-250418

(22)出願日 平成7年(1995)9月28日

(71)出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72)発明者 沼崎 俊一

神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

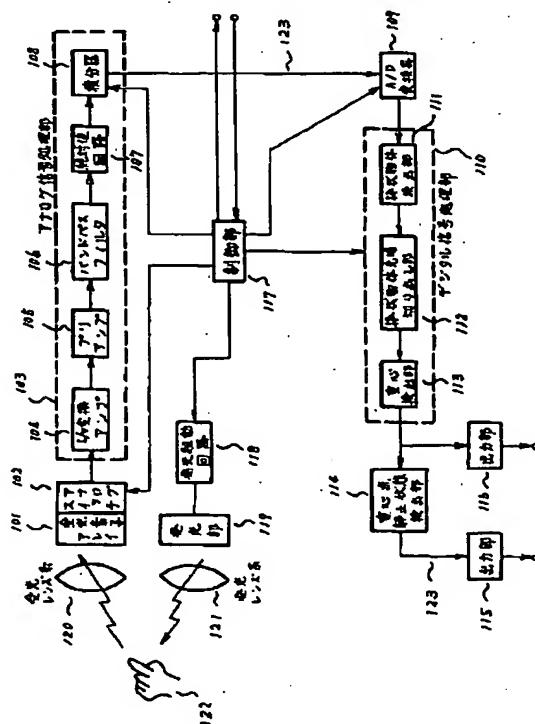
(74) 代理人 弁理士 外川 英明

(54) 【発明の名称】 情報入力装置および画像処理方法

(57) 【要約】

【課題】特に、ブックパソコンやPDAなどの携帯型情報機器において、簡単な操作入力手段を提供する。

【解決手段】変調光を発光し、その変調光の物体による反射光を受光部アレイで捕らえ、アナログ信号処理によって変調光成分だけ取り出すことによって、反射物体の形状の画像が得られる。反射物体を、指を伸ばした手と限定すれば、それ程数の多くない受光部アレイであっても、粗い画像であっても、わずかな指の動きで画素値が変化するため、受光画像の指先部分の重心を求めることで十分な分解能の2次元情報が得られ、ポインティングなどに利用できる。



BEST AVAILABLE COPY

【特許請求の範囲】

【請求項1】変調光を出力するための発光手段と、この発光手段によって出力された変調光であって、物体に反射した前記変調光を受けるための受光手段と、この受光手段で受けた変調光の成分を取り出すための手段と、この手段で取り出した変調光の成分の大きさを求めるための測定手段とを有することを特徴とする情報入力装置。

【請求項2】変調光を出力するための発光手段と、この発光手段によって出力された変調光であって、物体に反射した前記変調光を受けるための受光手段と、この受光手段で受けた変調光の成分を取り出すための手段と、

この手段で取り出した変調光の成分の大きさを求めるための測定手段と、

この測定手段で求めた変調光の成分の大きさをもとに、入力情報とすべき特徴量を検出するための特徴量検出手段と、

この特徴量検出手段で検出した特徴量をもとに、操作すべき電子機器を制御する手段とを有することを特徴とする情報入力装置。

【請求項3】棒状物体を含む画像から、先端の棒状物体の一部を構成する画素の画素値の合計がある条件を満たすように、いくつかの画素の画素値を修正し、修正された前記画素値とともに、前記棒状物体の一部の重心を求める、

求めた前記棒状物体の一部の重心をもとに、棒状物体の先端位置求めることを特徴とする画像処理方法。

【請求項4】変調光を出力するための発光手段と、この発光手段によって出力された変調光であって、物体に反射した前記変調光を受けるための受光手段と、この受光手段で受けた変調光の成分を取り出すための手段と、

この手段で取り出した変調光の成分の大きさを求めるための測定手段と、

この測定手段で求めた変調光の成分の大きさをもとに、前記物体の画像を生成する手段と、

この手段で生成された画像であって、棒状物体を含む画像から先端の棒状物体の一部を構成する画素の画素値の合計がある条件を満たすように、いくつかの画素の画素値を修正する修正手段と、

この修正手段によって修正された前記画素値とともに、前記棒状物体の一部の重心を求める重心検出手段とを有することを特徴とする情報入力装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、電子機器などを操作するための情報入力装置に関し、また、入力された画像から重心を抽出する画像処理方法に関する。

【0002】

【従来の技術】従来、パーソナルコンピュータなどに対する、文字情報以外の情報入力にはマウスなどのポインティングデバイスが多く用いられている。ポインティングデバイスは、コンピュータに位置情報を簡単に入力するための装置で、様々な形態の装置がある。最もポビュラーなのがマウスである。これは、装置の中に、自由な方向に回転できるように支持された球が入っていて、マウスを机上に置くと球の1点が机に接するようになっている。マウスを前後左右に動かすことによって、中の球がマウス本体に対して回転し、その回転量を2つの方向で機械的もしくは光学的に検出し、コンピュータに送っている。コンピュータ側では、送られてきた移動量の情報から、画面上のカーソルを動かす方向と距離を決定し、カーソルを再表示させる。

【0003】マウス以外にも、タッチパッド、トラックボール、小型ジョイスティック、タッチパネル、ペンデバイスなどさまざまなポインティングデバイスが、電子機器の入力装置として、実現されている。

【0004】

【発明が解決しようとする課題】マウスは簡単に位置情報を入力することができるが、いくつかの欠点を持っている。まず、マウスを操作するためには机上のようなある程度の広さを持った平らな面が必要である。従って、平らな面を確保できないような場所、状態での使用は難しい。

【0005】例えば、携帯パソコンを会社や自宅から持ち出して、屋外で使用しようとするときは、マウスを操作できる平面があるところを探さなくては快適に使用することができない。

【0006】例えば、電車や車の中で座っていて、膝の上にパソコンを置き使用するのには不便である。また、単に社内、自宅内で持ち歩くときでも、マウスごと移動させるのは面倒である。また、机の上で使うときでも、ごちゃごちゃと書類が散乱していれば、まず机の上を整理する必要がある。

【0007】現在はマウス以外のポインティングデバイスを搭載した携帯パソコンも増えている。マウスの替わりにトラックボールやタッチパッド、小型のジョイスティックなどを利用する。トラックボールは、ボールを指で動かし、その回転方向・量をカーソル移動方向・量に変換し入力する。タッチパッドは様々な方式があるが、指先でパッドに触ることで、その位置を検出できるようになっている。通常は、パッドをなでるように指を動かすと、その移動方向・量をカーソル移動方向・量に変換して入力する。

【0008】小型のジョイスティックは、キーボードの「G」と「H」と「B」の間の隙間に小型のレバー状のものがついており、これを任意方向に傾けると、その傾き方向と大きさをカーソル移動方向・量に変換して入力

する、方法を探るものが製品化されている。

【0009】マウス、トラックボール、タッチパッド、小型ジョイスティックなどによるポインティングは、基本的に、

(1) 現在のカーソル位置の把握

(2) カーソルの移動目標位置の決定

(3) 移動方向・量の確認

(4) 移動方向・量の入力

というプロセスに従って行われる。ここで(1) (2)の順序は逆のこともありうる。また、(3) (4)のプロセスは1回だけで終了するものではなく、カーソル移動中も、常に移動方向・量の修正を続けている。

【0010】ここで、操作者の手の動きに注目してみると、マウスと、それ以外のポインティングデバイスの間では違いがある。マウスを用いてポインティングする場合は、マウスの位置と、カーソルの位置はほぼ対応している。もっとも、マウスをいったん持ち上げて、移動させればその時点でマウス位置とカーソル位置の関係は変わるし、カーソルの速度を加速させる機能の付いたマウスドライバを用いた場合も、マウス位置とカーソル位置の関係は完全には対応しない。しかし、短い操作単位で考えれば、マウスの位置と、カーソルの位置はほぼ対応していると言つていい。つまり、カーソルを動かすときにカーソルの動きと同じようにマウスを持っている手を動かせばよい。

【0011】これに比べて、トラックボール、タッチパッド、小型ジョイスティックの場合は、カーソルの動きと手の動きは対応していない。トラックボールは、ボールの回転方向・量をカーソル移動方向・量に対応づけているが、ある程度以上の距離をカーソル移動する場合は、ボールを何回か回すことになる。つまり手の動きは往復運動をする。タッチパネルも同様である。小型ジョイスティックの場合は、レバーの傾き方向・量をカーソル移動方向・量に対応づけるため、カーソルの動きと手の動きは似ていない。つまり、カーソルが等速で動いているとき、手は静止しているし、カーソル位置を微調整するときは、レバーを小刻みに動かすことになる。

【0012】これらの違いは、上記(3) (4)のプロセスの繰り返しにおいて現れる。(3) (4)の繰り返しでは、まず大まかなカーソル移動を行い、次第に細かな移動を制御するという順序をたどることが多い。大まかなカーソル移動において、マウスを使う場合は、マウスの移動量とカーソルの移動量の比をだいたいの感覚で掴んでおけば、かなり高速に行うことができる。それに比べて、トラックボールやタッチパッドの場合は、移動距離が大きい場合は、ボールを何度も回す、あるいはパッドを何度もかざる、という動作をしなくてはならない。また小型ジョイスティックの場合は、レバーを傾けて、カーソルを動かしながら、カーソルの動きを監視し、目標に近づいたら、レバーの傾きをゆるめるという

動作をしなくてはならない。細かな移動を制御する場合、マウスを使用するときは、カーソルを動かしたい方向に動かしたいだけ、マウスを動かせばよい。タッチパッドの場合も同様であるが、指とパッドの摩擦であまり細かい移動をスムーズに行なうことは難しい。また細かい移動をスムーズにするために、指とパッドが軽く触れるように操作すると、時として指がパッドから離れてしまい、クリック入力を誤ってしまうこともある。また小型ジョイスティックの場合は、細かく移動する度に、レバーをちょっと倒してすぐ戻す、という動作を繰り返さなくてはならず、時間がかかるてしまう。概して、操作性はマウスの方が優れているが携帯性に劣るし、携帯性に勝るタッチパッド、トラックボール、小型ジョイスティックは操作性に劣る、と言うことができる。

【0013】またマウスを用いずにポインティングするために、透明タッチパネルを使うこともできる。指で押したときにその位置を検出できる透明タッチパネルを、ディスプレイ上に重ねれば、直接画面上の任意の点を指示できる環境が実現できる。これは前記(1)～(4)のプロセスのほとんどを省略でき、

(1) ポインティングしたい位置を確認する

(2) その場所を指でタッチする

というシンプルなプロセスで実現できる。しかし、問題点も多い。まず指で直接画面を押すので、押すときには画面のその部分、および周辺が見にくくなる。また、そのために指の大きさより細かい精度でのポインティングは原理的に難しい。さらに直接触るので、画面が汚れやすいという問題もある。ドラッグをするために画面の上を指でなぞるというのは汚れの面でも、指先の摩擦の感触の面でも抵抗がある。ドラッグに関しては、1回のタッチでアイコンなどを拾い、2回目のタッチで別の場所へドロップすることもできるが、指がタッチパネルに触れていないときはその位置を検出できないので、アイコンを引きずって見せるという、視覚的な効果は出せない。実際、銀行のATMの操作パネルなど、大きなボタンを押すものによく用いられ、一部のワープロに採用されている以外は、携帯パソコンなどへの搭載はあまりなされていない。

【0014】タッチパネルではペンデバイスと併用するものもあり、パーソナルな電子機器への入力方法としてはこちらの方がポピュラーである。これはペンコンピュータやPDA(パーソナル・デジタル・アシスタント)の入力インターフェースとして、数多く商品化している。ペンを併用すると、指でタッチするのとは異なり、細かいポインティングも可能であるし、画面が汚れることがなく、ドラッグなどの作業も快適に行える。さらに、手書き文字などの入力が好適に行えるという特徴も持っている。しかし、これはペンを手に持たなくてはならない、という欠点がある。ペンだけで全ての操作が行えるようなPDAには良いが、携帯パソコンのようにキーボ

ードを併用する場合は、いちいち画面をポインティングするためにペンを持つというのは面倒である。またPDAなどにおいても、文字入力をするならともかく、画面の操作だけで終わるような、情報検索のような作業をするのにもいちいちペンを取り出さなくてはならないのは操作上の負担になる。また、ペンはなくなる可能性があり、またなくなないように線でつながったペンは使いづらい。

【0015】全く別のアプローチとして、画像処理によって手の形、位置を認識し、入力手段とする方法も試みられている。これが実現すれば、画面を指差すだけでポインティングが可能なので、カーソルの動きと指の動きはマウス程度に対応し、タッチパッド、トラックボール、小型ジョイスティックなどに比べて、操作しやすい。加えて、マウスのように平面を必要としないので、どこでも操作することができる。またタッチパネルのように画面を汚すこともなく、ペンデバイスのようなものをいちいち手に取る必要もない。処理が簡単なものとして、色マーカーを付けた手を撮像し、マーカー部分を検出する方法が考えられるが、この方法は動作測定・分析装置などの特定用途で製品化されることが多い。携帯パソコンのポインティング手段として応用できなくはないが、画像処理ハードウエアが高価である、マーカーと同じ色が服などにあると誤動作する、マーカーを付けるのが面倒などといった問題点があり、適しているとは考えにくい。何も付けない裸の手の画像から手の形状や指先の位置を認識して、ポインティングに利用するシステムは、研究レベルではいくつか試作されている。しかしこれらは、手の画像を背景から容易に切り出せるような環境でしか適切に動作させることができず、複雑でかつ動きのある背景の下で手の形を切り出すのは非常に難しい。

【0016】以上述べたように、現在実現されているポインティング技術はどれも一長一短であり、特に携帯パソコンなどを屋外に持ち出して自由に使用したいとき、それに適したポインティング手段が十分あるとは言いかない。本発明において、携帯パソコンに使用して、その自由度を十分発揮できるようなポインティング方法及び装置を実現する。

【0017】

【課題を解決するための手段】本発明は、変調光を出力するための発光手段と、この発光手段によって出力された変調光であって、物体に反射した前記変調光を受けるための受光手段と、この受光手段で受けた変調光の成分を取り出すための手段と、この手段で取り出した変調光の成分の大きさを求めるための測定手段とを有することを特徴とする情報入力装置である。

【0018】また、変調光を出力するための発光手段と、この発光手段によって出力された変調光であって、物体に反射した前記変調光を受けるための受光手段と、

この受光手段で受けた変調光の成分を取り出すための手段と、この手段で取り出した変調光の成分の大きさを求めるための測定手段と、この測定手段で求めた変調光の成分の大きさをもとに、入力情報とすべき特徴量を検出するための特徴量検出手段と、この特徴量検出手段で検出した特徴量をもとに、操作すべき電子機器を制御する手段とを有することを特徴とする情報入力装置である。

【0019】さらに、棒状物体を含む画像から、先端の棒状物体の一部を構成する画素の画素値の合計がある条件を満たすように、いくつかの画素の画素値を修正し、修正された前記画素値とともに、前記棒状物体の一部の重心を求め、求めた前記棒状物体の一部の重心をもとに、棒状物体の先端位置求めることを特徴とする画像処理方法である。

【0020】そして、変調光を出力するための発光手段と、この発光手段によって出力された変調光であって、物体に反射した前記変調光を受けるための受光手段と、この受光手段で受けた変調光の成分を取り出すための手段と、この手段で取り出した変調光の成分の大きさを求めるための測定手段と、この測定手段で求めた変調光の成分の大きさをもとに、前記物体の画像を生成する手段と、この手段で生成された画像であって、棒状物体を含む画像から先端の棒状物体の一部を構成する画素の画素値の合計がある条件を満たすように、いくつかの画素の画素値を修正する修正手段と、この修正手段によって修正された前記画素値とともに、前記棒状物体の一部の重心を求める重心検出手段とを有することを特徴とする情報入力装置である。

【0021】このように、本発明では、装置を付けない裸の手指で、画面を指さし、動かすことで、電子機器の操作ができる環境を提供する。例えば、携帯パソコンを使用しているときには、キーボードから手を少し浮かせて画面を指さし、動かすことで、画面内のカーソルを制御できる。また、指を指文字のように決められた動きに従って動かすことでコマンドを発生させることもできる。

【0022】しかし、CCDエリアイメージセンサなどで撮像した手の画像を認識する、といった手法は取らない。本発明では、この環境を実現するために、発光部と受光部を組み合わせる。すなわち、操作者の手に光を当て、手指からの反射光を受光部で受ける。この受光部は複数用意しておき、例えばアレイ状に並べておく。受光部の前方には受光レンズ系があり、これにより各受光部は指向性を持ち、ある方向のある範囲からの光を受ける。ある受光部は手からの反射光を受け、ある受光部はその指向性のある方向に手がなかったため反射光を受けない。

【0023】また、各々の受光部の指向性のある方向にある手指の距離や、そこが手指の端なのか真ん中なのか、などによって、その受光量が異なる。この系での受

光量は、光を反射する物体の距離の4乗に反比例するので、手の後ろに体があるような場合でも、手からの反射か、体からの反射かは、その受光量によって十分弁別できる。従って、これらの受光部からの出力は、手指の位置や形を反映している。従って、これらの出力を処理することによって指先の位置を求め、ポインティング情報として利用することができる。

【0024】しかし、受光部には発光部の手指による反射光だけでなく、太陽や電灯などの光も入射してくる。これらの影響を排し、受光部からの光だけを取り出さなくては正確な情報は得られない。そのため、発光部は変調光を出力する。受光部あるいはそれ以降のアナログ処理部において、その変調成分だけを取り出すことで、正確に手指からの反射光だけを取り出すことができる。

【0025】このようにして取り出した出力は、各々の受光量を画素値とすれば、画像のように扱うことができる。従って、画像処理によって、その出力を処理し、指先の情報を得ることができる。しかしこの画像はCCDエリアイメージセンサなどで撮像した画像とは違い、近くにある手の形状だけを表し、後ろの方にある背景は画像には現れない。操作するときは指を1本伸ばしていると仮定すると、画像の中から、ほぼ上方に向かって伸びている、細長い部分を取り出し、その先端部などの特徴点を取り出せばそれが指先であると判断することができる。これは、粗い画像でしかないが、指先が（画素の大きさに比べて）微少に動いたことが、各画素の画素値の微少な変化に反映されるので、これらを反映する特徴量を用いれば、画素数以上のポインティングが可能になる。

【0026】また、マウスのボタンに対応する機能も様々な方法で実現できる。上述の画像の処理によりカーソル位置が求められたとき、そのカーソルの静止状態を検出するようにしておき、カーソルが一定時間静止したときにクリックが行われたようにすることができる。また、指の受光センサからの距離に反映した特徴量を計算し、その変化を観ることで、「押す」（近づいて、遠ざかる）という動作を検出することができる。従って、「押す」という動作をクリック動作に関連づけることもできる。さらに、他にボタンを用意しておきこれをマウスのボタンと同様に利用することもできる。この場合は右手でカーソルを動かし、左手でボタンを押すというよう両手で操作できる環境では便利である。

【0027】以上のようにして、裸の手指を画面に向けて指さすだけで、ポインティングできる環境が実現できる。またその指先の動きを追跡することで、ジェスチャーによるコマンド入力も実現できる。さらに、画像処理をさらに工夫すれば、指先位置だけでなく手の形や向き、距離などの情報も得ることができ、さらに複雑な情報を入力することも可能である。

【0028】

【発明の実施の形態】これより本発明について、実施例を用いて詳細に説明する。図1は本発明の実施例の全体構成を表している。使用者は発光部の光が当たって受光部に返っていくような位置に手を差し出し、その指先を動かすことによって、2次元情報を入力できる。操作のイメージを図2に示す。ここでは、使用者はブック型などの小型パソコンを使用している。使用者は、キーボードを操作している手を、キーボードよりやや情報を持ち上げ（図では分かりづらいが、画面に触れてはいない）、指先を動かすことによって、画面内のカーソルの動きを制御することができる。この図ではカーソルが左上のメニュー上にあり、このメニューが反転表示されている。本体とディスプレイのつなぎ目あたり、キーボードのすこし上にある2つの丸い物体はレンズであり、それぞれの奥に後述する発光部、受光部が配置されている。

【0029】図1に戻って、その詳細を説明する。全体の構成は大まかに分けて、発光部119と発光駆動回路118、受光素子アレイ101、アナログスイッチ102、アナログ信号処理部103、A/D変換器109、デジタル信号処理部110、およびそれらを制御する制御部117から成る。

【0030】各構成要素について詳しく説明する。まず、LEDなどの発光部119が発光して操作者の手122を照らす。この発光部は発光駆動回路118によって変調発光される。この変調光を図3の301に示す。変調光は操作者の手122に反射して、その一部が受光光学系120を経て、受光部に捉えられる。ここでは受光部はフォトダイオードをアレイ状に並べた受光素子アレイ101を用いる。この受光素子アレイからの複数の出力をアナログスイッチ102で選択してアナログ信号処理部103に導く。アナログ信号処理部103はI/V変換アンプ104、プリアンプ105、バンドパスフィルタ106、絶対値回路107、積分器108で構成される。フォトダイオードは光により電流を発生させるため、まず電流-電圧(I/V)変換を行い、プリアンプ105でもう1段増幅する。受光素子アレイ101にはLEDの手による反射光だけでなく、太陽光や白熱灯、蛍光灯のような電灯の光などが入射し、これはノイズとなる。バンドパスフィルタ106はこれらのノイズ成分を遮断し、変調光成分だけを通過させる。バンドパスフィルタ106の出力を絶対値回路107、積分器108を通過させることにより、手によって反射した変調光の強さに応じた信号123を取り出すことができる。

【0031】ここで、アナログ信号処理部103の各部での信号を図3に表す。まず、発光部からは、ある周波数のパルス光がある時間間隔をおいて、繰り返し出力される。アナログ信号処理部103の入力部においては、太陽光のような定常的な外光、蛍光灯のような周期的な外光の成分がノイズとして乗っているので302のようになっている。バンドパスフィルタ106を通過させる

と、ノイズは除かれ、変調されたパルス成分だけが取り出され、303のようになる。これを絶対値回路に通した波形が304、積分器の出力が305である。受光量の大きさが変われば、303の波形の振幅が変わるのを、305における積分後の信号レベルが変わることで、変調されたパルス光の反射光の大きさだけが忠実に取り出せる。

【0032】この図ではある周波数のパルスを複数発光して一つの受光部の出力を得ているが、S/Nの性能を許容できるならば、代わりに短パルスを用いても良い。この変調光の強さに応じた信号をA/D変換器109でデジタルデータに変換する。アナログスイッチ102の切り替えが一巡し、全ての受光部からの出力をA/D変換し終わつたところで、1フレーム分のデータができる。以下の説明において、一つの受光素子に対して、その出力がアナログ処理されて（ノイズ成分を除去されて）、A/D変換された後の値を、その受光部の反射受光量と呼ぶことにする。

【0033】ここで受光部の数はそれ程多くなくても良い。同じ受光センサアレイであるCCDなどは数十万画素程度のフォトダイオードを集積化しているが、ここではそれほどの数は必要としない。それよりも、各々の受光部で反射受光量を正確に捉えられることが大事である。本発明の受光系の目的は、対象物を正確に撮像することにあるのではなく、指先の位置を検出するのに十分な情報を捉えることがある。本実施例では 20×26 に受光部を並べている。1つの受光部当たり、操作空間内で約10mm四方の範囲からの光を捉える。従って、全部で $200\text{mm} \times 260\text{mm}$ の空間をカバーする。操作者はこの空間内に指先が入るようにして動かすことによって、カーソルを動かすことができる。図4は、この受光素子アレイが捉える領域の一部とそこにある操作者の手を表したものである。マスの1つが、1つの受光素子が捉える領域である。10mmという数字は、指の幅を超えないできるだけ大きな値であり、任意の受光部に対し、適当な指の位置で、その捉える領域が指先で占められるようできる。図4ではマス403がこれに相当する。この条件が満たされないと、後述するように、指の動きに伴ってカーソルがスムーズに動かない。そして反射受光量は、その捉える領域を手指がどのくらい占めているか、またその距離、によって決まる。

【0034】従って、捉える領域を手指が全て占めていれば、反射受光量は大きいし、半分程度占めていれば反射受光量はその半分になることになる。マス403に対応する受光素子の反射受光量は大きく、マス404に対応する受光素子の反射受光量は403の半分程度となる。図5は図4の各マスに対応する反射受光量を垂直軸にとって表したものである。各受光部の反射受光量を画素値とすれば、これは粗い濃淡画像になるが、これを以下で反射受光画像と呼ぶことにする。

【0035】次にこのデータを処理して指先を求めるのがデジタル処理部110である。通常の操作状態では、操作者の手は指示する指（一般的には人差し指）を上に伸ばした形になっており、その上方には変調光を反射する物体は存在しない。厳密には、操作者の頭などが変調光を反射しているわけであるが、前述したとおり、反射受光量は距離の4乗に反比例するので、手までの距離の4倍の距離に頭があつても、その反射受光量は256分の1になつてしまい、影響はほとんどない。手の距離の2倍だったとしても、16分の1になり、あるしきい値を設ければ十分に取り除ける。

【0036】図6に示すように、反射受光画像606を上部から水平にスキャンしていくと、あるところまでは画素値が0に近い画素ばかりであるが（601～602）、あるところまで来ると、ごく一部だけに大きな画素値が現れる。そのような水平ラインがいくつか続き、しかも大きな画素値が現れる画素の横方向の位置がほぼ同じであれば（603～605）、細長い物体が縦に存在していると分かる。つまり伸ばした指が存在していれば、それを検出することができる。次にその指先部分からポインティングに利用する特徴点を抽出する。

【0037】一般的には指の先端を使うのが良いと思われがちであるが、ここでは1画素の大きさが大きく、反射受光画像の指先付近は、図7のようになっているので、単純に指先が存在している座標701を使うとポインティングの精度は画素の数と同じになつてしま（ここでは、 20×26 ポイント）。ちなみにこの図に現れている格子は、そのひとマスからの反射光が1つの受光部に入射することを表している。ここでは実際の画素の数よりはるかに細かい精度で指先の位置を検出する。

【0038】本実施例では、指先端ではなくて、図のX印702の位置を画素以上の分解能で求め、これを特徴点としてポインティングに用いる。これは指先の一部分の（像の）重心である。この重心を粗い反射受光画像からどのように抽出するかを説明する。

【0039】デジタル信号処理部110は棒状物体検出部111、棒状物体先端切り出し部112および重心検出部113から成る。棒状物体検出部111では、縦方向に伸びた棒状の物体、すなわち操作者の手の中で上方に伸びた指（通常は人差し指）を検出する。次に、棒状物体先端切り出し部112では、縦方向に伸びた棒状の物体（人差し指）の上部先端を切り出し、後の重心検出部113でその切り出した部分の重心を求める。この時、画素単位で切り出してしまうと、画素が粗いため切り出し部分の画素値合計が大きく変動してしまい、検出された重心の精度が悪くなる。従って、棒状物体先端切り出し部112では、切り出し部分の画素値合計がある条件を満足するように一部画素値を修正しながら切り出す。

【0040】さらに各部について詳しく説明する。ま

ず、棒状物体検出部111について述べる。ここでは、使用者が操作をするために指を出したとき、指示する指（通常は人差し指）が、反射受光画像の上方向に伸びている、と仮定している。従って、この仮定に合わない形状で手を出すと（例えば、グーで差すなど）正常に検出できない恐れはある。

【0041】まず、反射受光画像606を、上方から水平方向にスキャンしていく。指先が、反射受光画像の上端になければ、通常は上の方は、1ライン全ての画素がほぼ0に近い画素値を持つ（601～602）。水平方向の1ラインの中で、部分的に画素値が大きくなるところがあった場合（603）、その位置と、そのラインを覚えておく。これを数ライン繰り返し（604、605）、部分的に画素値が大きくなるところが数ライン

$$TH = \sum_{x=1}^n \sum_{y=1}^{k-1} P(x, y) + a \cdot \sum_{x=1}^n P(x, k) \quad (0 < a \leq 1)$$

となるような、k、aを求める。次にaを用いて、P(1, k)～P(n, k)の画素値を修正する。つまり、古い画素値をP'(1, k)～P'(n, k)とし

$$P(x, k) = a \cdot P(x, k)$$

とする。この修正されたP(1, k)～P(n, k)を含む、P(1, 1)～P(n, k)が切り出された部分803となる。803の上部2行は802の上部2行と同じであり、803の最下行は802の3行目のそれぞれの画素値にaを乗じて修正されている。次に重心検出部において、切り出されたP(1, 1)～P(n, k)を使って、重心(xx, yy)は次の式を満たすように決定される。

【0045】

【数3】

$$xx = \frac{\sum_{x=1}^n \sum_{y=1}^k x \cdot P(x, y)}{\sum_{x=1}^n \sum_{y=1}^k P(x, y)}$$

【0046】

【数4】

$$yy = \frac{\sum_{x=1}^n \sum_{y=1}^k y \cdot P(x, y)}{\sum_{x=1}^n \sum_{y=1}^k P(x, y)}$$

【0047】ここで求められた(xx, yy)は、図4の絵の中の伸びた指のうち上方からある長さだけ切り取ったもの（切り取る長さがしきい値THと関連する）

の中で一致すれば、そこに縦方向に伸びた棒状の物体（例えば、人差し指）があると判定できる。そして、伸びた指がありそうな所を、大きめに囲う。この部分画像について、さらに以下の処理を行う。

【0042】ここからは、棒状物体先端切り出し部112での処理である。図8の801がもともとの反射受光画像であり、802は棒状物体検出部111で、この中に指先があると判定された部分画像である。一番左上の画素値をP(1, 1)とし右上をP(n, 1)、左下をP(1, n)、右下をP(n, n)とする。ここで、あるしきい値THを設定しておき、

【0043】

【数1】

たとき、

【0044】

【数2】

(x = 1, 2, ..., n)

の像の重心とほぼ一致する。ここでは通常のTV画像に比べて画素が粗いため、切り取る長さを一定に保つために、上記のような修正が必要になる。ここでは、各画素の値P(x, y)が精度良く求まれば求まるほど、(xx, yy)の値も精度良く求められる。

【0048】ここで、P(1, k)～P(n, k)の値を上のように修正した理由を述べる。もし、P(1, k)～P(n, k)の値を修正しないで、(xx, yy)を計算したらどうなるかを図9を用いて考えてみる。今、仮に水平ラインで一部だけ大きい画素値が現れるところから、3画素分下がったところまで（斜線部分）計算に用いるとすると、指が(a)から(b)のように移動したとき、指の位置変化に比べて、(xx, yy)の位置の変化が大きくなってしまう。逆に図10(a)と(b)は指の位置変化に比べて(xx, yy)の位置の変化が小さくなってしまう。

【0049】つまり、実際の指の位置と計算された(xx, yy)の値に、線形性がなくなってしまう。従って、上述のように、まず重心を計算するときの全画素の画素値の合計が一定になるように補正をしてから重心の計算をしなくてはならない。

【0050】前述したが、任意の受光部に対し、適当な指の位置で、その捉える領域が指先で占められるような条件を満たさなくてはならない理由についてここで述べる。もし、画素の捕らえる領域の幅が、指の幅より小さくなってしまうと、図11のように、ある範囲においては指を動かしているにも関わらず、(xx, yy)は動

かない、という状況が発生してしまう。そうすると、上述したのと同様に、実際の指の位置と (x, y) の値に線形性がなくなってしまう。

【0051】以上で求められた座標 (x, y) を使って、画面のカーソルを制御する。 (x, y) と画面内のカーソル位置を対応づけておけば、指を差し出したときその場所に対応した場所にカーソルが表示される。また指を動かせばカーソルも動く。指を引っ込めれば、カーソルは消える。ただこれだけでは、マウスのボタンに対応するものがない。マウスのボタンを押す（クリックする）代わりにここでは、カーソルを静止させる。すなわち、重心点静止状態検出部 114 が重心点の動き（カーソル位置に対応している）を監視しており、ある一定時間での移動距離がしきい値以下だった場合、カーソルが静止状態であったと判定し、クリック信号 123 を出力する。プルダウンメニューの中から 1 つを選択する作業を考えてみる。まず指を差し出すことでカーソルが現れるので、まず操作者は指を動かして、カーソルをメニューの項目の所へ動かす。カーソルをメニュー項目の上に移動させて一定時間静止させると、プルダウンメニューが現れる。ここでさらに指を下方に動かすと、メニューを反転させながらカーソルが動く。希望のメニューの所でまたカーソルを静止させると、そのメニュー命令が起動される。マウスなどを用いるときはマウスを操作していないときは常にカーソルは静止しているため、この方法では誤操作が多いように一見思えるかもしれない。

【0052】しかし、本実施例においては、指を差し出してないときはカーソルは現れていないし、指を出すときはある目的を持っているため（あのメニューを選択しよう、など）、目標の場所にカーソルが到達する前にカーソルが静止することはあまりない。従って、本実施例においては、カーソルを静止させることでクリック動作を代用することは自然である。

【0053】次に、デジタル信号処理部における処理の流れを図 18～20 に示したフローチャートを用いて説明する。ここでは、上で述べたよりもやや細かい工夫が施されているところもあるが、それらも含めて以下で説明する。このフローチャートをそのままプログラムにすれば、デジタル信号処理部をソフトウェアで実現することができる。もちろん、ハードウェアで構成することも可能であるし、ソフトウェア、ハードウェア両方で構成されることもあり得る。

【0054】図 19 は棒状物体検出部の処理を表している。ここでは既に述べたように、反射受光画像を水平方向にスキャンしながら、部分的に画素値が大きくなるところを見つけ、これが複数ラインで、近い位置に現れると、棒状物体があると判断する。この部分的に画素値が大きくなるところ（以下で部分画素列と呼ぶ）、および部分画素列の複数の集まりで、まだ棒状物体とは判断さ

れてはいないが棒状物体になる可能性のあるものを棒状物体候補と呼ぶ。水平方向にスキャンしていくと、棒状物体候補が見つかり、スキャンが進むにつれその中から棒状物体が検出される。

【0055】まず、この棒状物体候補の情報を格納しておく構造体を複数用意する（1901）。ここでは構造体名は `infoStick[n]` とする（`[n]` がついているのは複数あるから）。複数用意するのは、棒状物体候補は必ずしも 1 つではなく複数現れる可能性もあるからである。この構造体は 5 つのメンバーを持ち、そのうち 2 つは配列である。図 21 の 2102 にも示すように、メンバーは、

`ystart` : 棒状物体が始まったラインの `y` 座標、
`xstart[i]` : 棒状物体候補の上から `i` ライン目（最上段は 0 ライン）の部分画素列の始まりの `x` 座標、
`xend[i]` : 棒状物体候補の上から `i` ライン目（最上段は 0 ライン）の部分画素列の終わりの `x` 座標、
`length` : 棒状物体候補の縦方向の長さ（部分画素列の数）、

`status` : 現在の状態（STICK、NOSTICK、EXTENDED の 3 つの値を取る。詳細は後述。）、をそれぞれ表す。始まりの `x` 座標、終わりの `x` 座標は、複数のラインにそれぞれあるので、これらは配列となっている。

【0056】以下で処理を説明していくが、わかりやすいように図 21 を併用しながら述べる。2101 は反射受光画像を表しており、各画素の色の濃さは画素値の大きさを表している。また上部と左部の数字はそれぞれ `x` 座標、`y` 座標を示している。2102 には前述の構造体の概要を表す。右側の 2103～2105 は処理の途中での構造体の内容である。

【0057】水平ラインをスキャンしていく、条件を満たす（ここでは、画素値が `s` 以上の画素が、`c` 個以上連続しない）部分画素列が見つかると（1902）、これまでの棒状物体候補に統合するか（1907）、新規に棒状物体候補にする（1906）。1 番最初に見つかった部分画素列は必ず新規棒状物体候補になる。例えば、図 21 では、`y = 3` のラインをスキャンしているときに、始めて部分画素列が見つかり、これを新規棒状物体候補として登録する（2103）。

【0058】このラインの `y` 座標、すなわち 3 を `infoStick[0].ystart` に入れる。部分画素列の長さは 1 なので、`infoStick[0].xstart[0]`、`infoStick[0].xend[0]` ともに 6 を代入する。

【0059】新規なので、`infoStick[0].length` は 1 とし、`infoStick[0].status` には EXTENDED を格納する（新規登録でも EXTENDED を使う。理由は後述）。

【0060】`y = 4` のラインのスキャンで見つかった部

分画素列は今度は、既にある棒状物体候補とつながる。つながるかどうかの判断は、棒状物体候補の最後のラインの部分画素列と新たな部分画素列のx方向の位置関係で決める。

【0061】この場合はinfoStick[0].xstart[0]=6, infoStick[0].xend[0]=6である。新たな部分画素列のx座標は5~7なので、これらはつながっていると判断する。その結果、新たな部分画素列につながっていると判断された棒状物体候補の情報を更新する(2104)。

【0062】新たな部分画素列の(x方向の)位置を下式で代入し、infoStick[0].xstart[1]=5, infoStick[0].xend[1]=7。

【0063】棒状物体候補の長さを1増やす。
infoStick[0].length=2
さらに、y=5のラインで見つかった部分画素列を先の棒状物体候補にさらにつなげると、2105のようになる。ここでdを3とすると、1908の条件式が成り立ち、棒状物体が決定され、次の処理へ移る(1909)。この条件が満たされるまで繰り返され、条件が満たされる前に1画面分のスキャンが終了してしまったら、棒状物体は検出できなかったことになる。

【0064】上の説明では、infoStick[n].statusの使い方の説明が不十分なので、次にこれについて説明する。これは、STICK, NOSTICK, EXTENDEDの3つの値を取りうる。

【0065】NOSTICKは、その構造体が棒状物体候補でないことを表す(つまり、一番はじめは全ての構造体のstatusメンバーはNOSTICKになっている)。STICKはその構造体が棒状物体候補であることを表す。

【0066】EXTENDEDの場合も棒状物体候補であるがこの値の用い方は少し異なる。新たな水平ラインのスキャンを始めるとき、全ての棒状物体候補はSTICKを持つ。1ラインのスキャンの中で新たに見つけられた部分画素列とつながった棒状物体候補はEXTENDEDに変わる。1ラインのスキャンが終了したとき、まだSTICKになったままの棒状物体候補は、そのライン上のどの部分画素列ともつながっていないことになり、棒状物体候補でなくなったと考える。

【0067】従って、1ラインのスキャンが終わると、STICKになったままの棒状物体候補はNOSTICKに変え、EXTENDEDを持つ棒状物体候補だけに再びSTICKを与える。この処理により図22の2201のように、浮いた部分画素列(ノイズである場合も考えられる)は、いったんは棒状物体候補としてinfoStick[0]に登録されるが(2203)、画素列がとぎれた時点で棒状物体候補から外される(2204、実際には1903の処理で削除される)。

【0068】本来の指先部分はinfoStick[1]で登録されている。棒状物体候補から外すためにstatusをNOSTICKに変えているだけである。これで次のスキャン時の1905での判断時には、STICKを持つ構造体だけを見るので、NOSTICKになった構造体は無視される。またこのように一度使われた構造体も、NOSTICKになれば、再び新規棒状物体の登録に用いることができメモリを節約できる。

【0069】このような処理を行うことでノイズから受ける影響を少なくすることができる。1905においては、STICKを持つ構造体に対してだけ、つながるかどうかの判断を行う。また、新規棒状物体候補を登録するときにはNOSTICKを持つ構造体を探し、そこに新しい情報を格納する。

【0070】図20は棒状物体先端切り出し部、および重心検出部の処理を表す。図19の1908での条件が満たされると、図20の2001に処理が移る。2002では、棒状物体の先端を切り出すためのパラメータTHを求めている。このしきい値は固定でも良いが、2001のように棒状物体の先端部の大きさや画素値によって決めてても良い。ここで挙げた式を用いると、指の太さが違ったり、距離が違ったりして、画素値が変動しても、切り出した先端部の縦横方向の長さの差がそれ程大きくならない。棒の太さ(1ラインの画素値合計)。0ライン目は先端なので太さを表さないことが多いので計算から除く)と1画素の最大値との比を求めて、TH(切り出し部分の全画素値合計)と最大画素値の比がこれの2乗になるようにしている。これ以外の計算方法でTHを求めることが可能である。

【0071】2003では、2002で求められたTHを用いて、式1で述べたa, kを求める。式1では矩形領域で計算しているが、ここでは、各ラインの部分画素列だけを用いて計算している。これにより、矩形内の棒状物体と関係のないところにノイズがあるような場合の影響を小さくできる(例えば図23のように指が斜め方向を向いているとき、2301のようなノイズの影響を受けない)。2003で求めたaをkライン目の各画素値に乘する。2002から2004までの処理が棒状物体先端切り出し部である。a, kと修正された画素値を用いて、2005において重心を計算する。基本的には、kライン目までの部分画素列の中の、各画素値とx座標の積を全て足してTH(これは各画素値の合計に等しい)で割ったものが重心のx座標であり、各画素値とy座標の積を全て足してTH(これは各画素値の合計に等しい)で割ったものが重心のy座標である。ここでは、xsum[i]が分かっているので、これにy座標(infoStick[n].ystart+i)を掛けたものの積をTHで割って重心のy座標を求めている(もちろん、xsum[k]は修正された画素値で再計算される)。xpsum[i]はiライン目の部分画素

列の各画素値とそのx座標の積を全て足したものであり、2005に示した式で重心のx座標が求められる。

【0072】以上述べた、実施例により、ノートパソコン、PDAなど携帯型情報機器により好適な入力環境、すなわち指を差し出して動かすだけでカーソル制御を行える環境を実現することができた。

【0073】次に、この実施例を基本とした、いくつかの別の実施例について説明する。ここで、さらに低コスト化、小型化のための実施例を述べる。ここでは受光センサとアナログスイッチ、アナログ信号処理部、A/D変換部が集積されて1チップ化されている。

【0074】このICの概要を図12に示す。ここではフォトダイオード1201とMOSスイッチ1202で1画素分を構成し、縦横に配列する。縦に並んだ画素のMOS出力をまとめ、縦1列分の出力をさらに後段のMOSスイッチ1203を介してまとめて後段のアナログ信号処理回路1206に導いている。

【0075】そして、後段のMOSスイッチ1203をX駆動回路1204で制御し、各画素のMOSスイッチのゲートは横1列で共通化し、これらをY駆動回路1205で制御している。つまり、X駆動回路1204が制御する制御線のうち1本だけをONにし、Y駆動回路1205が制御する制御線のうち1本だけをONにすれば、ただ1つのフォトダイオードの出力が後段のアナログ処理回路1206に導かれる。

【0076】アナログ信号処理回路1206はI/V変換アンプ1207、プリアンプ1208、バンドパスフィルタ1209、絶対値回路1210、積分回路1211からなっており、これらの働きは前述の実施例とほぼ同じである。A/D変換部1212で変換されたデジタルデータが出力端子1214より出力される。制御部1213は外部からの制御信号1215をもとに、X駆動回路1204、Y駆動回路1205の制御、A/D変換器1212のタイミング制御、積分回路1211のリセット（放電）動作制御などを行う。

【0077】このようにセンサ部を集積化することにより、装置全体を小型化することができるだけでなく、対ノイズ性能を向上させることができる。このICを用いれば、あとはデジタル信号処理を行うIC（CPU、DSP、専用LSIなど）と組み合わせ、全体でも数個のICで本入力装置を構成することができる。また、デジタル信号処理を、本入力装置を組み込む携帯パソコン（あるいはPDAなど）のCPUに行わせることにすれば、センサICとインターフェース用のICのみで本入力装置を構成することもでき、さらに低コスト化、小型化できる。

【0078】また別のセンサICの例として図13のようなものもある。これはアナログ信号処理回路を複数持ち、受光素子アレイを分けて同時に処理している。アレイ型に並んだ受光素子を左右2つに分け、それぞれにア

ナログ信号処理回路を設けている。アナログ信号処理部は図12と同様に、I/V変換アンプ、プリアンプ、バンドパスフィルタ、絶対値回路、積分器から成っている。またX駆動回路1301から出ているゲート信号は1本につき、2つに分けた受光素子アレイのそれぞれの1つずつの後段MOSスイッチを同時に制御する。

【0079】つまり、X駆動回路1301、Y駆動回路1302それが1本づつの制御線をONにすれば、左半分にある1つの受光素子の出力が第1のアナログ信号処理回路1304に導かれ、右半分にある1つの受光素子の出力が第2のアナログ信号処理回路1303に導かれる。

【0080】この2つのアナログ信号処理回路が所定の出力を出したとき、それぞれの出力をサンプルホールド回路1305、1306でラッチし、これをマルチプレクサ1307に入力し、時分割でA/D変換する。

【0081】制御部1309は図12で説明したほかに、サンプルホールド回路1305、1306、マルチプレクサ1307のコントロールも行う。このようにアナログ信号処理回路を複数持ち、並列処理させることの一番の効果は省電力化である。1回の発光で2つの受光素子出力を処理できるため、発光回数が半分になる。これは発光に必要な電力を半分に減らさせることを意味している。

【0082】図13では、2分割した例を述べたが、もちろんいくつに分割して並行処理させてもかまわないし、それだけ発光電流を節約できる。このようなフォトダイオードをアレイ状にした受光センサとしては、似たようなものに、CCDエリアイメージセンサがある。受光するだけであれば、広く普及しているCCDエリアイメージセンサを用いることもできるが、これはある一定時間チャージした電荷を30分の1秒毎に順番に取り出す構造をしており、本実施例のように変調された光を捉える場合、その変調成分を取り出すことはできない。それに、前述したように伸ばした指の位置を求めるというだけの目的であれば、CCDエリアイメージセンサのように多くのフォトダイオードを集積化する必要もなく、またCCDエリアイメージセンサほど高速動作を要求されないので省電力化でも有利である。

【0083】次に、デジタル信号処理が異なる例を説明する。先の実施例では指先の画像から、その先端部分を切り出し、その重心を求めただけであった。重心は指の左右上下方向の移動に伴い変化する特徴量であるが、ここでは、さらに奥行き方向の動き（受光系に近づく／遠ざかる）を反映した特徴量を捕らえる。これまでの実施例では、2次元情報を入力するのみであったが、奥行き方向の情報を捉えることができれば、2次元情報+ α あるいは3次元情報を入力することができる。

【0084】例えば、奥行き方向の動きを監視しており、ボタンを押すように、少し近づいて少し戻るという

奥行き方向の動きを検出した場合に、マウスのクリックに対応する信号を発生させることができる。これにより、指でカーソルを動かし、「押す」動作をすることで、メニューを選択させたりすることができる。

【0085】また、厳密に指までの距離を入力することはできないが、相対的に「近づいた」「遠ざかった」という検出はできるので、表示でのフィードバックと組み合わせれば3次元位置情報を入力することができる。

【0086】受光量は反射物体までの距離の4乗に反比例することは前述した。従って、指が奥行き方向に動くと、指からの反射光を受光している受光素子の受光量が増減する。また1つの受光素子の出力だけでなく、指の占める画素の数も増える。この様子を図14に示す。

(a) → (b) のように指が近づいてくると、手指の形の反射受光画像全体に対する相対的な大きさが大きくなる。

【0087】従って、3行目の水平1ライン1401について1.1画素程度の範囲を指が占めていたのに対し、(b)では1.5画素程度の範囲を指が占めており(1402)、3行目1ラインの画素値合計が大きくなっていることが分かる。また占める範囲が広がるだけでなく、指の距離が近くなるので、捉える領域全てを指が占めている1403、1404のような画素も、(b)の方がその画素値は大きくなっている。前述のように受光量は反射物体までの距離の4乗に反比例するので、例えば距離が10cmから9cmに近くなったとすると、画素値はほぼ1.5倍になっている。

【0088】この図から、水平1ラインでの画素値合計の大きさが、指までの距離を反映していることが分かる。この実施例でのデジタル信号処理部(図15全体)は、棒状物体検出部1501、水平1ライン総画素値検出部1502、棒状物体先端切り出しパラメータ演算部1503、棒状物体先端切り出し部1504、重心検出部1505から成る。まず、反射受光画像1506が入力されると、棒状物体検出部1501が指先を検出する。これは図1の棒状物体検出部111と働きは同じである。次に棒状物体検出部1501で検出された部分画像1507の各水平ラインの画素値合計を演算する。

【0089】棒状物体先端切り出しパラメータ演算部1503では、この画素値合計から、棒状物体先端切り出し部1504で用いるしきい値THを計算する。棒状物体先端切り出し部では、棒状物体先端切り出しパラメータ演算部で計算されたTHを受け取り(1508)、図1のそれと同様に、先端部を切り出す。重心検出部1505もその働きは図1のそれと同じである。そしてこの重心データ1509と切り出しパラメータ1510の両方が特徴量として出力される。

【0090】このデジタル信号処理部の効果は距離に対応した信号が得られるだけではない。THが一定だと、指が近いほど、すなわち水平1ラインの総画素値が大き

くなると切り出された部分の垂直方向の1ラインあたりの画素値合計は小さくなる。ポインティングの精度を高く保つためには、切り出された部分の1ラインあたりの画素値合計が縦でも横でも同じ程度が望ましい。

【0091】従って、水平1ラインあたりの画素値合計から、垂直1ラインあたりの総画素値も同程度になるようにTHを決める方が望ましい。また先に述べた、マウスのクリックとしての動作だけを行わせるのであれば、棒状物体先端切り出しパラメータ演算部1503の後に図示しないクリック動作検出部を設け、これが切り出しパラメータを監視しており、「押す」動作を検出したときにクリック信号を出力するようにすればよい。

【0092】「押す」だけでなく、様々な指先のジェスチャーをコマンドとして利用することも効果的である。図17に示す実施例では操作者の指の動きを認識して入力情報として利用する者である。例えばあるアイコンの上にカーソルを移動させて、チェックマークの形「#」に指を動かすと、そのアイコンに関連づけられたコマンドが起動する、または、円を描くとその中のオブジェクトが選択される、といった具合に利用することができる。

【0093】センサ部1701は図1の受光素子アレイ101とアナログスイッチ102に相当する。アナログ信号処理部1702、A/D変換器1703、デジタル信号処理部1704は図1のそれと働きは同じであり、重心データ1705を出力する。発光部もあるがここでは図示しない。重心点動き検出部1706は重心点の動きを監視しており、あらかじめ登録された動きが合った場合にそれに対応する信号1707を出力する。

【0094】次に、重心データとボタンの状態とを関連づけて、入力情報とする実施例について説明する。例えば、指を動かしてカーソルをあるメニューの上に動かしたところで、ボタンを押すとメニューが選択される、または、あるアイコンの上でボタンを押したまま、さらに指を動かすとアイコンを移動できるなど、マウスのボタンの同じように用いてクリックやドラッグなどの操作ができるようになる。

【0095】この実施例の構成は図16に示される。センサ部1601は図1の受光素子アレイ101とアナログスイッチ102に相当する。アナログ信号処理部1602、A/D変換器1603、デジタル信号処理部1604は図1のそれと働きは同じであり、重心データ1605を出力する。発光部もあるがここでは図示しない。ボタン1606は操作者の画面を差している手と反対の手で押しやすい所に配置されており、ボタン状態検出部1607が押されたかどうか、押され続けているかどうか、といったボタンの状態を検出し、その状態を表す信号1608を出力する。図示しないコンピュータ側ではこれら両者の出力を参考にして、カーソルを動かしたり、アイコンを起動したり、ドラッグしたりする。これ

により、指を静止させたり、「押す」動作をさせるよりは、高速にクリック操作などをすることができる。また、操作したという確実な実感を操作者に与えることができる。

【0096】これは、両手が自由に使えるような場合に有効である。腕時計型のPDAを左腕に装着して右手でカーソル操作したり、小型のPDAを左手を持って右手でカーソル操作するような場合は、片手でクリック操作ができる方が望ましい。また、ブックパソコンのような場合、このボタンをキーボードのキーで代用することもできる。

【0097】例えば、スペースバーを使った場合、通常はスペースの入力として働くようにしておき、指を差してカーソル表示させているときにスペースキーが押された場合は、クリック、ドラッグとして働くようになる。

【0098】また、少し違う使い方になるが、ボタンを押したときにだけこの情報入力装置が動作するようにすることもできる。すなわち、カーソル操作をしたいときはボタンを押しながら指を差して動かすことになる。こうすることにより、カーソル操作する意志がないのに操作領域に指が入ってしまい誤動作するのを防ぐことができる。また、そのため操作領域をキーボードのホームポジションのごく近いところに設定し、最小限の手の動きで操作領域に手が入るようにすることもできる。またボタンが押されたときだけ、発光部が発光するので、消費電力を節約することもできる。

【0099】最後に、2つ、この入力装置を備えた携帯情報機器のイメージを載せる。図18は片手で持てるPDAである。本入力装置は外側には発光、受光できるだけの窓があればよいので、スペースを有効に使える。ここでは、1801が発光部、受光部の窓である。窓の前で指1803を動かせば、画面内のカーソル1804の位置を制御できる。窓がこれだけ小さいので、残りのスペースを大きな表示部1802として使える。

【0100】図24は、本入力装置が付いた腕時計型のPDAである。やはり指2401の動きでカーソル2402を制御することができる。2403、2404が発光部、受光部の窓である。これだけ本体が小さいともはやペンを格納するスペースさえなく、本入力装置による入力の好適さは言うまでもないことである。また、目と表示装置の間からずらしたところに操作空間を配せば、指によって表示が見にくくなるということもない。

【0101】

【発明の効果】本発明により、マウスのように操作するための平面を必要としないポインティング環境を実現できる。また、マウスのように指の動きとカーソルの動きが一致するため、トラックボールやタッチパッドや小型ジョイスティックよりも素早く、正確にポインティングすることができる。またこれらに不向きなジェスチャー入力も実現できる。

【0102】また、タッチパネルを指で押す方式のように、画面が汚れることもなく、ペンデバイスのように、いちいち手に装置を持つ必要もない。これらの特徴を総合すると、携帯パソコンを操作するに当たり、既存のポインティング装置よりも快適な操作環境を実現することができる。

【0103】またPDAのようなさらに小型の電子機器に対する操作もより手軽に行えるようになる。さらにペンを収納するスペースもなく、指で触るには画面が小さすぎるような、腕時計ほどの小型のPDAを考えた場合、本発明は既存のポインティングデバイスに比べはるかに快適な環境を提供しうる。

【0104】また本発明の受光部は指先の位置を求めるのに必要最小限の受光アレイで実現している。従って、CCDのように高い集積度で受光アレイを作る必要がなく、低コストで生産することができる。

【図面の簡単な説明】

【図1】 本発明の構成を表す図である。

【図2】 本発明の情報入力装置を搭載した携帯型パソコンを操作しているイメージ図である。

【図3】 アナログ信号処理部の各部での信号を表す図である。

【図4】 受光素子アレイが捉える領域の一部と操作者の手の関係を表す図である。

【図5】 反射受光画像をその画素値を垂直軸に取り表したグラフを表す図である。

【図6】 反射受光画像の水平ラインをスキャンした様子を表す図である。

【図7】 本発明で検出する特徴量を説明する図である。

【図8】 反射受光画像と、指先が含まれる部分画像と、切り出された先端部画像を示す図である。

【図9】 棒状物体先端切り出し部がなかったときに、指先の動きと重心の動きに線形性がなくなる例の図である。

【図10】 棒状物体先端切り出し部がなかったときに、指先の動きと重心の動きに線形性がなくなる例の図である。

【図11】 1画素の捉える範囲が大きすぎる場合に、指先の動きと重心の動きの線形例がなくなる例の図である。

【図12】 受光素子アレイ、アナログスイッチ、アナログ信号処理部、A/D変換部を集積したICの構成を表す図である。

【図13】 アナログ信号処理を並列に行う、センサ部ICを示す図である。

【図14】 指までの距離が変わったときの受光画像の変化を説明する図である。

【図15】 指までの距離の変化を捉えるための、デジタル信号処理部の図である。

【図16】重心点とボタン状態を関連づける情報入力装置の構成の一部の図である。

【図17】指の動きをコマンドに変換できる情報入力装置の構成の一部の図である。

【図18】本入力装置を搭載した小型PDAの外観を表す図である。

【図19】棒状物体検出部の処理を表す図である。

【図20】棒状物体先端切り出し部および重心検出部の処理を表す図である。

【図21】棒状物体候補を格納する構造体の図である。

【図22】反射受光画像にノイズが乗っている場合の棒状物体候補の変遷の様子を表す図である。

【図23】棒状物体の先端近くにノイズが乗っている場合にその影響を除くことができることを説明する図である。

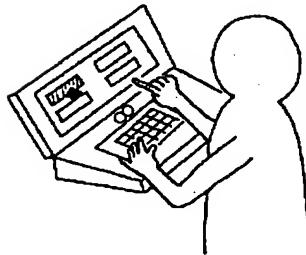
【図24】本入力装置を搭載した腕時計型のPDAの外観を表す図である。

【符号の説明】

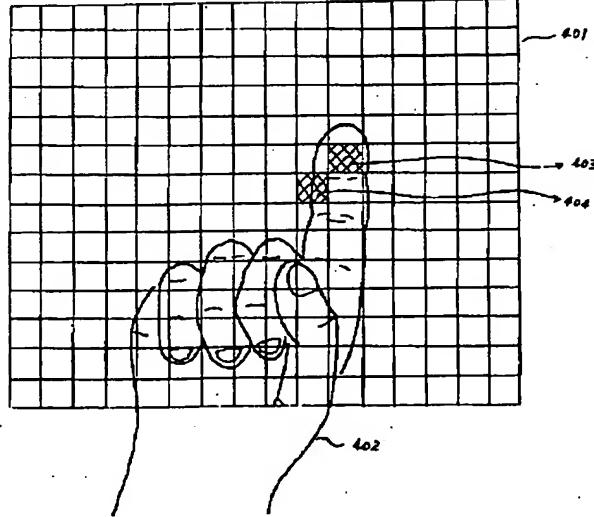
- 101 受光素子アレイ
- 102 アナログスイッチ
- 103 アナログ信号処理部
- 104 I/V変換アンプ
- 105 プリアンプ
- 106 バンドパスフィルタ

- 107 絶対値回路
- 108 積分器
- 109 A/D変換器
- 110 デジタル信号処理部
- 111 棒状物体検出部
- 112 棒状物体先端切り出し部
- 113 重心検出部
- 114 重心点静止状態検出部
- 117 制御部
- 118 発光駆動回路
- 119 発光部
- 1201 フォトダイオード
- 1202 前段MOSスイッチ
- 1203 後段MOSスイッチ
- 1204 X駆動回路
- 1205 Y駆動回路
- 1307 マルチプレクサ
- 1502 水平1ライン総画素演算部
- 1503 棒状物体先端切り出しパラメータ演算部
- 1606 ボタン
- 1607 ボタン状態検出部
- 1706 重心点動き検出部
- 2102 棒状物体候補を格納しておく構造体
- 2103~2105 棒状物体候補を格納する構造体の値の変化の例

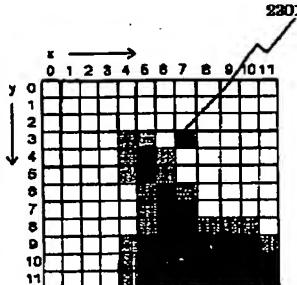
【図2】



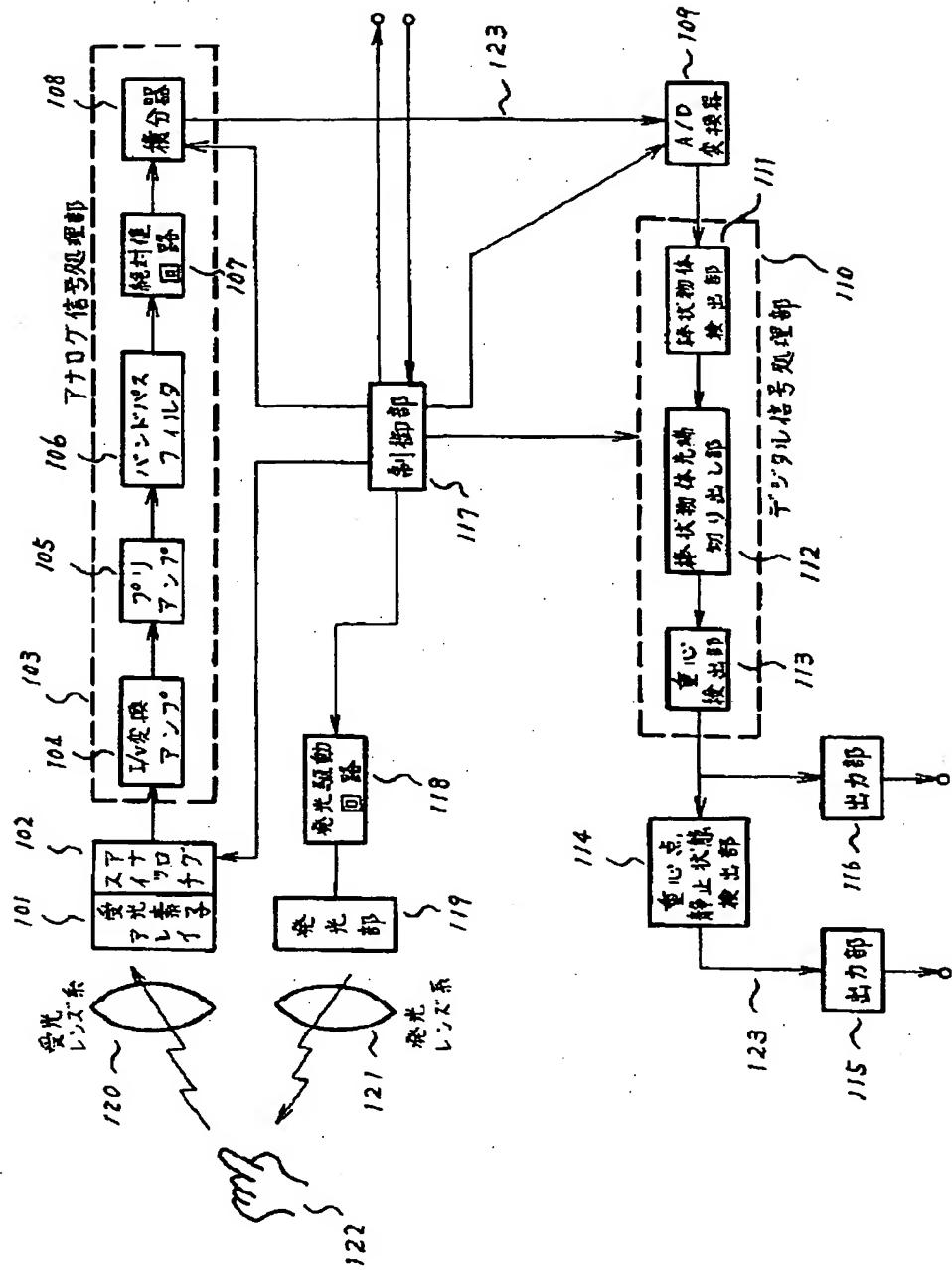
【図4】



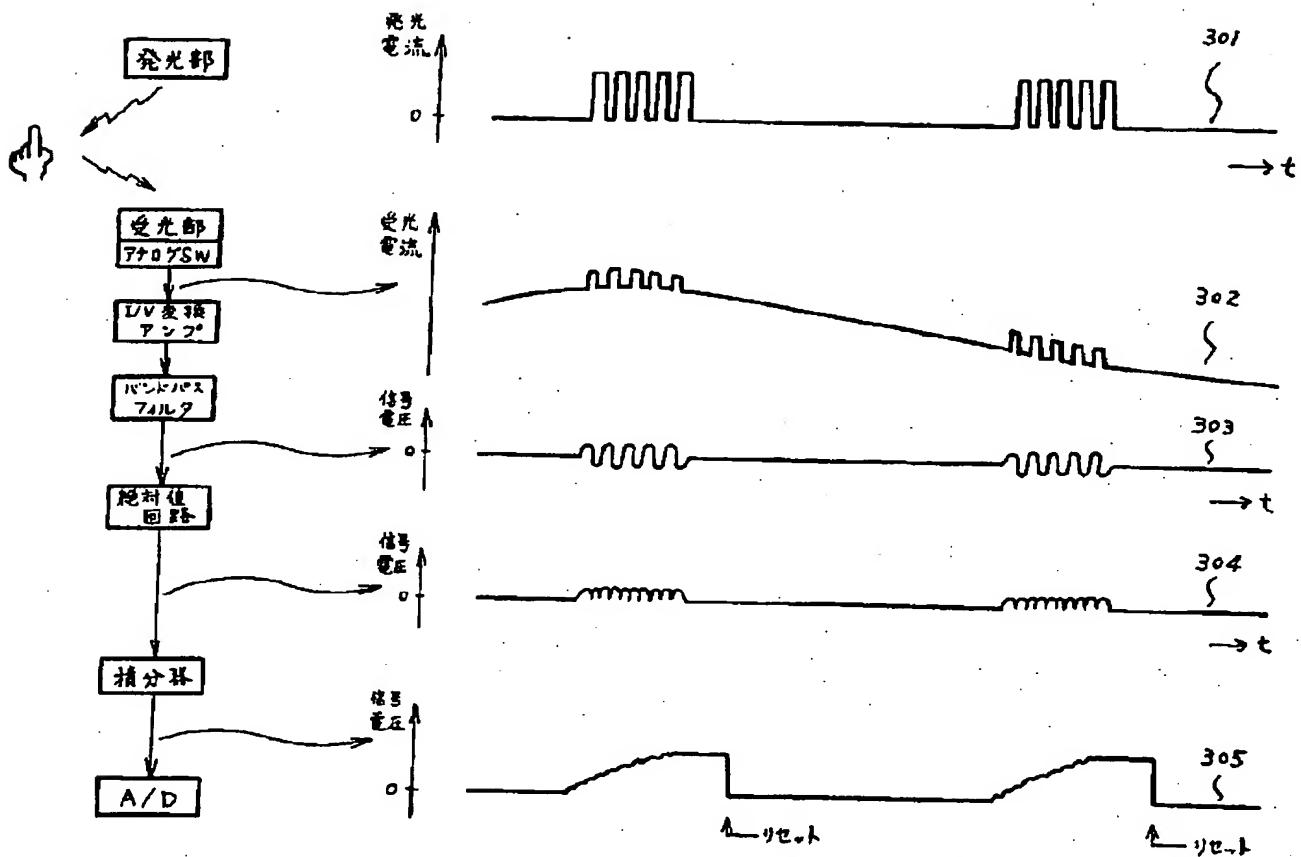
【図23】



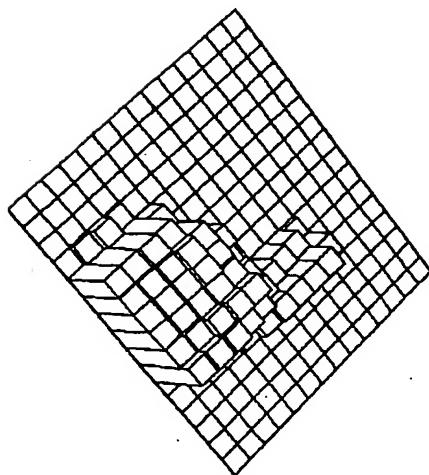
〔図1〕



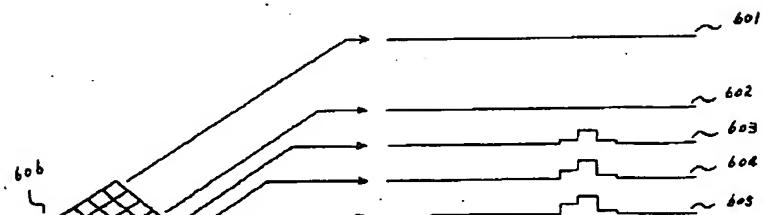
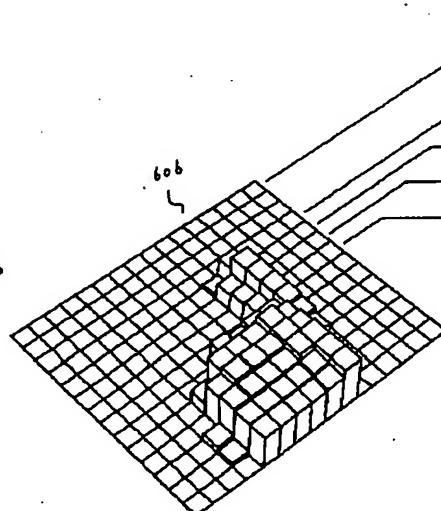
【図3】



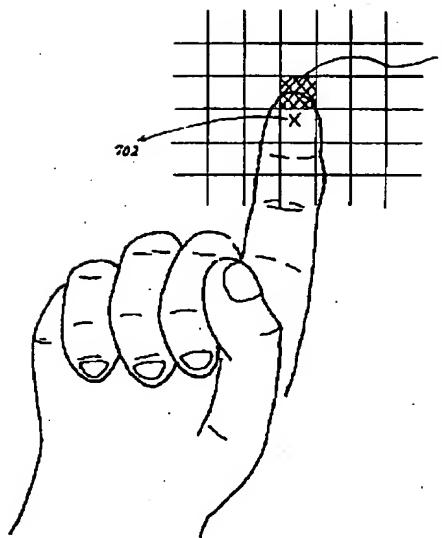
【図5】



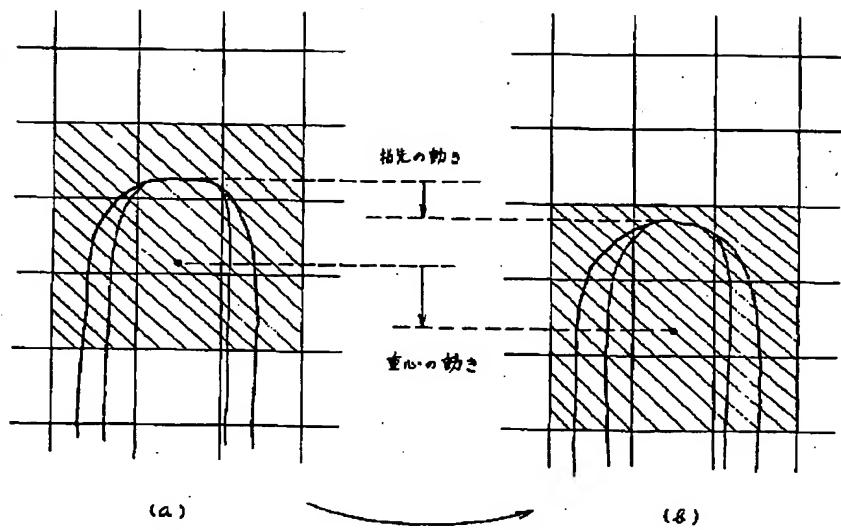
【図6】



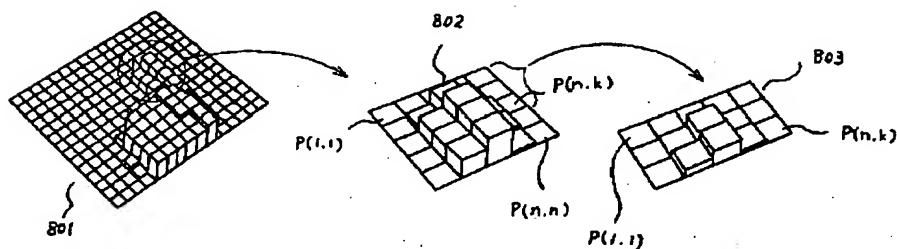
【図7】



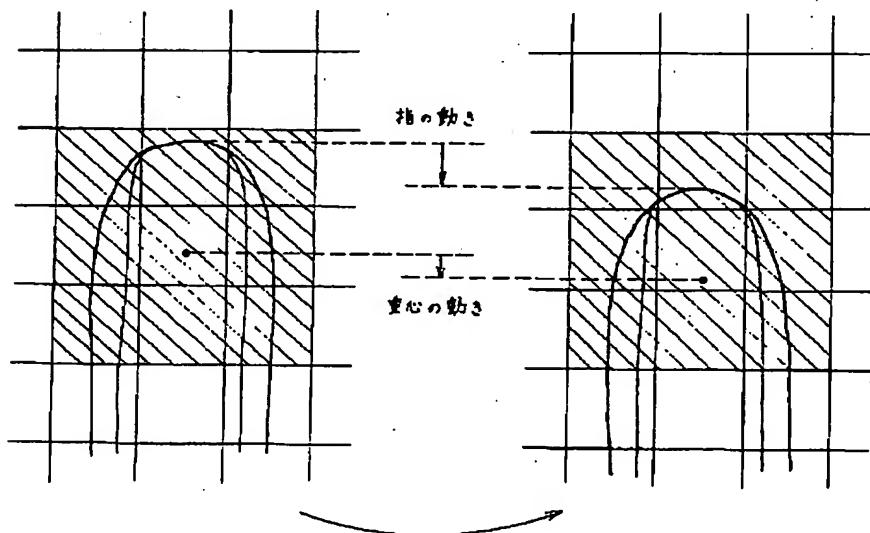
【図9】



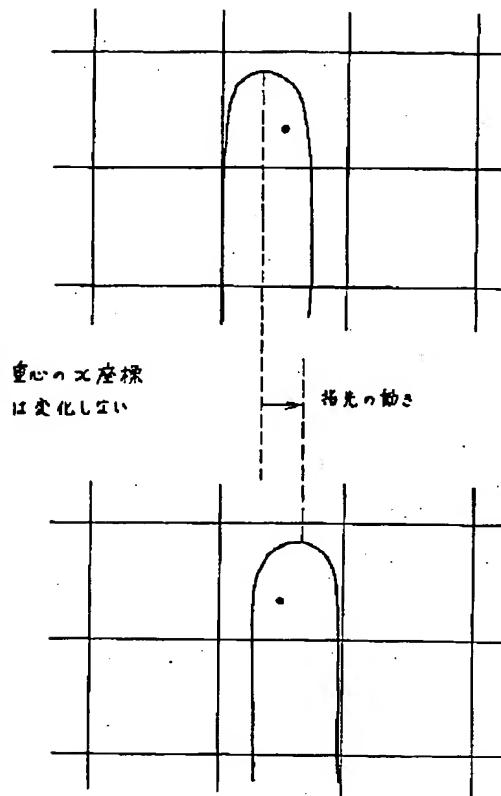
【図8】



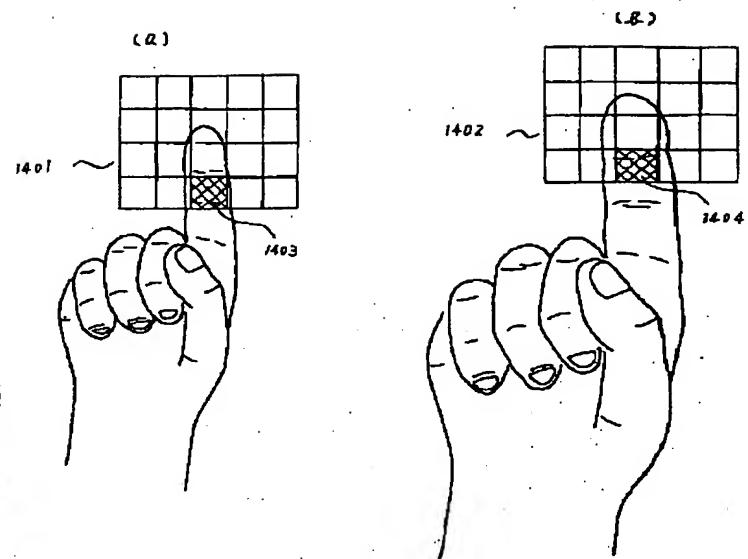
【図10】



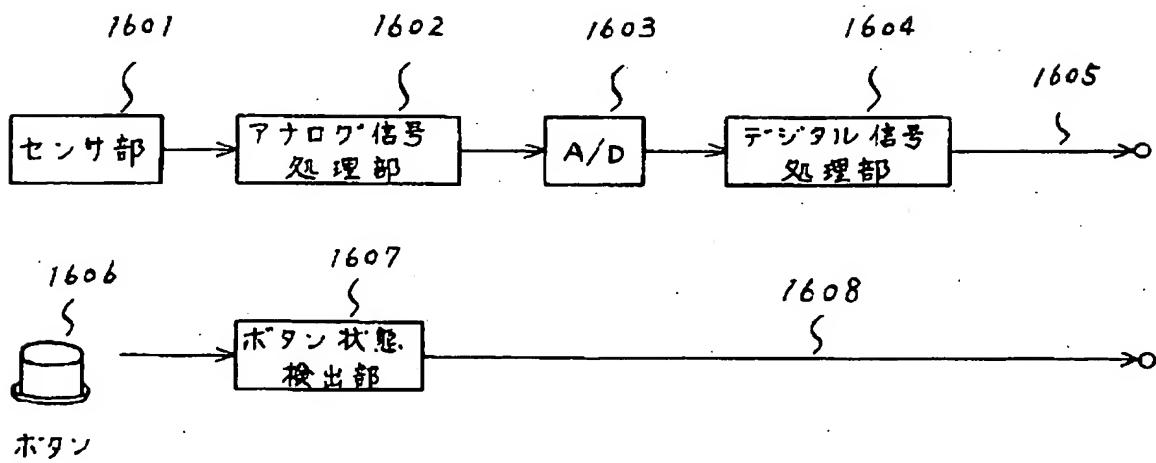
【図11】



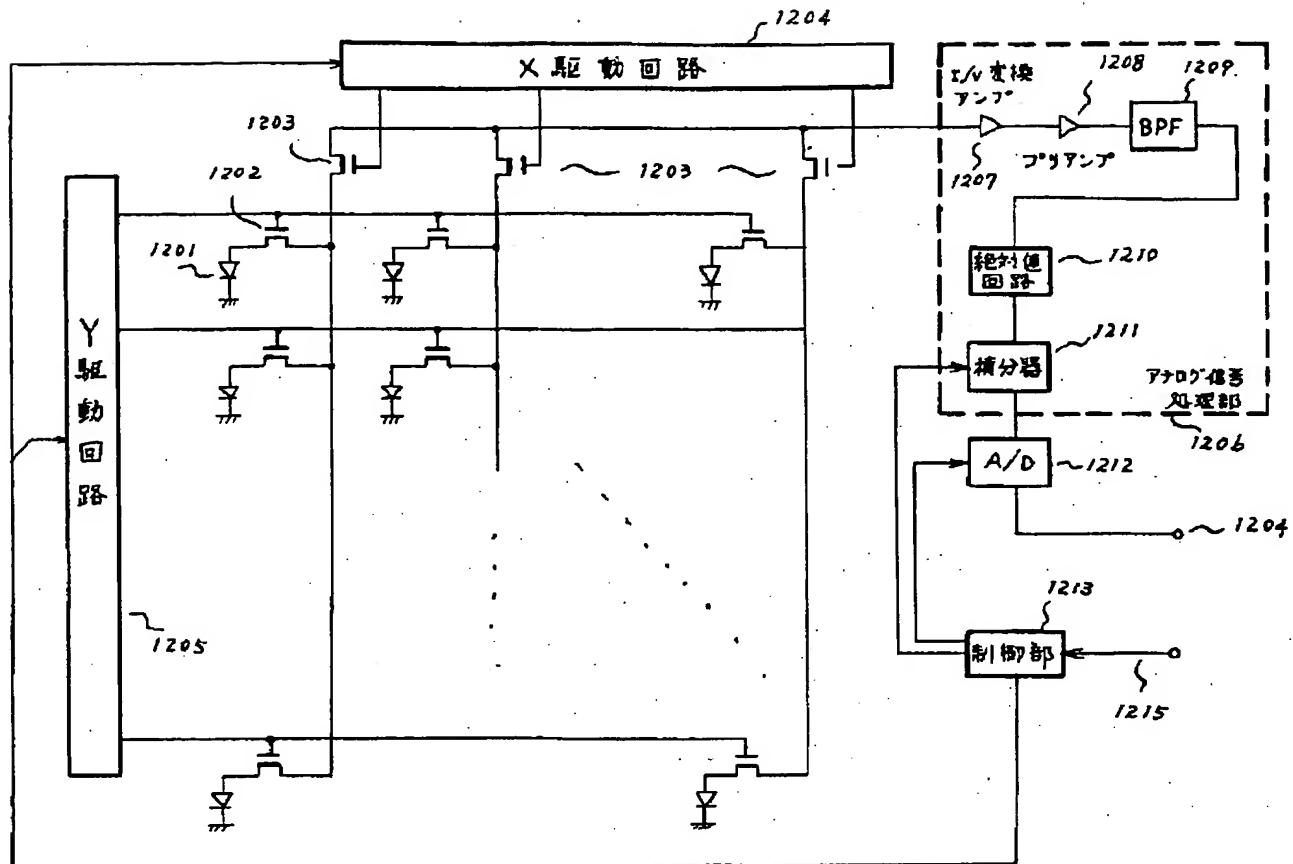
【図14】



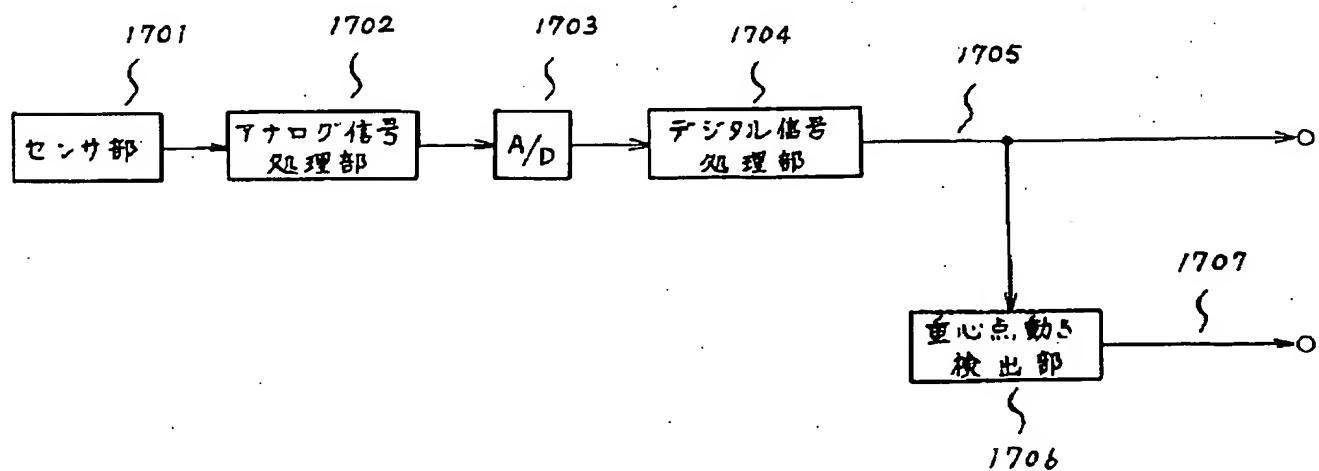
【図16】



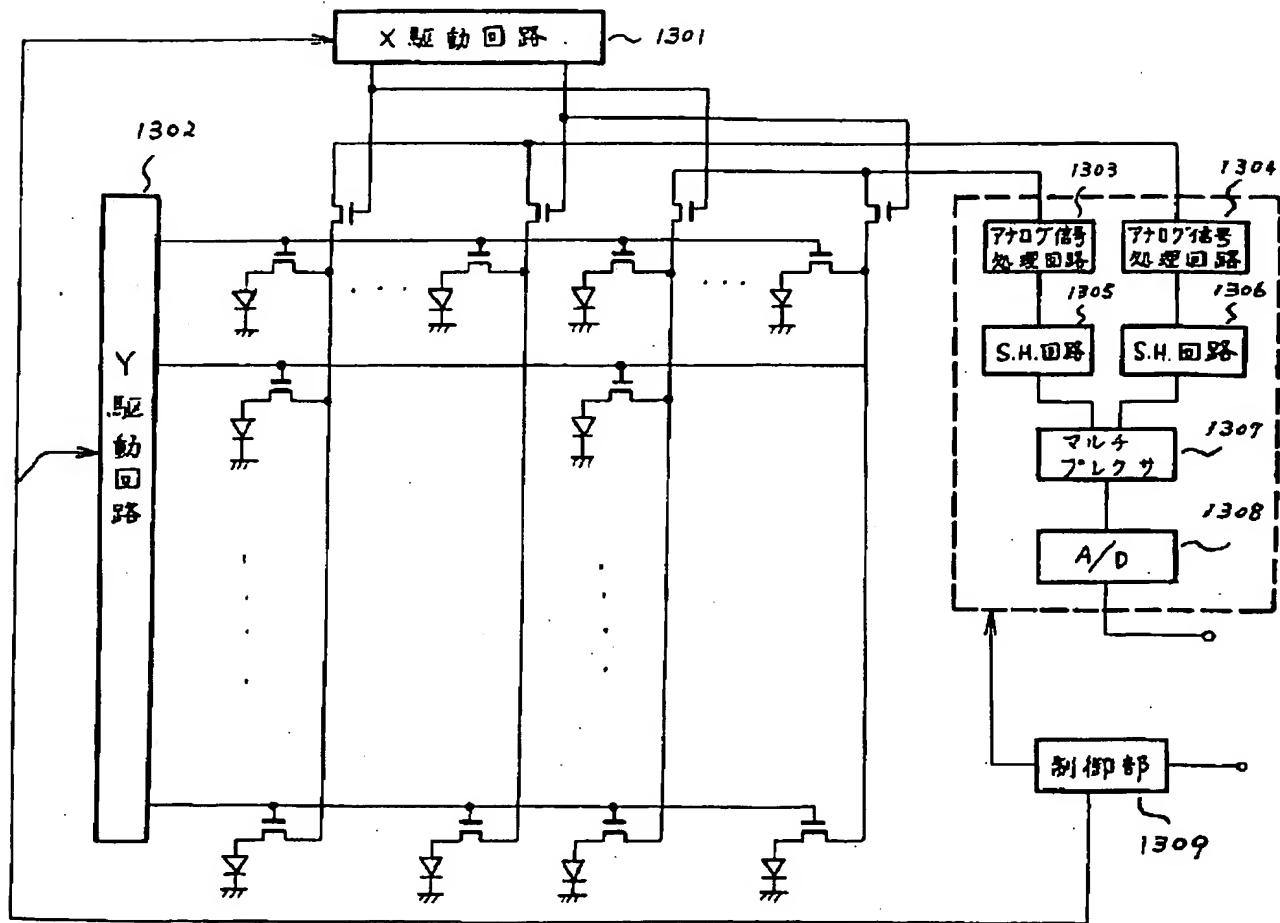
【図12】



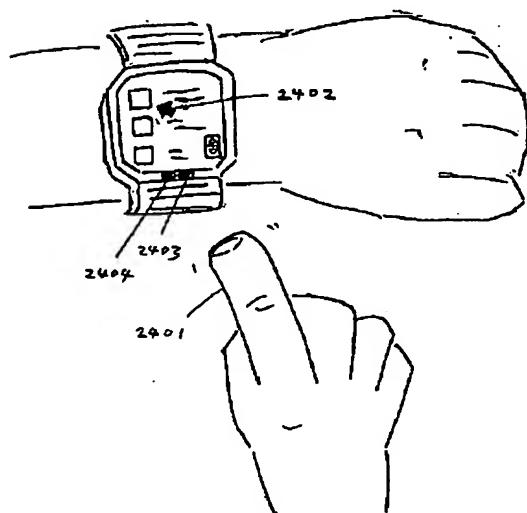
【図17】



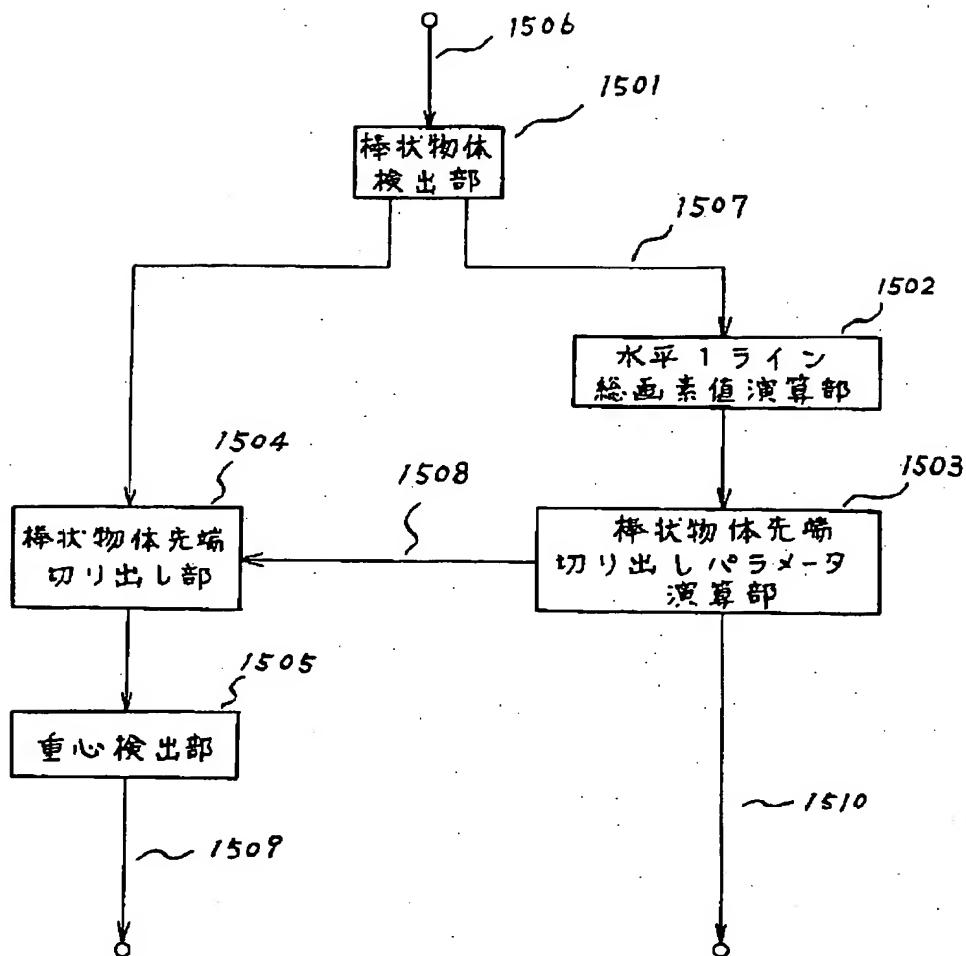
【図13】



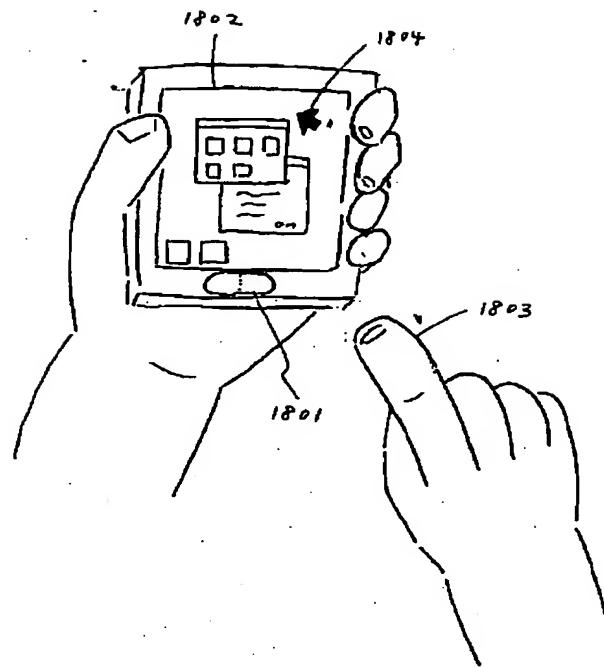
【図24】



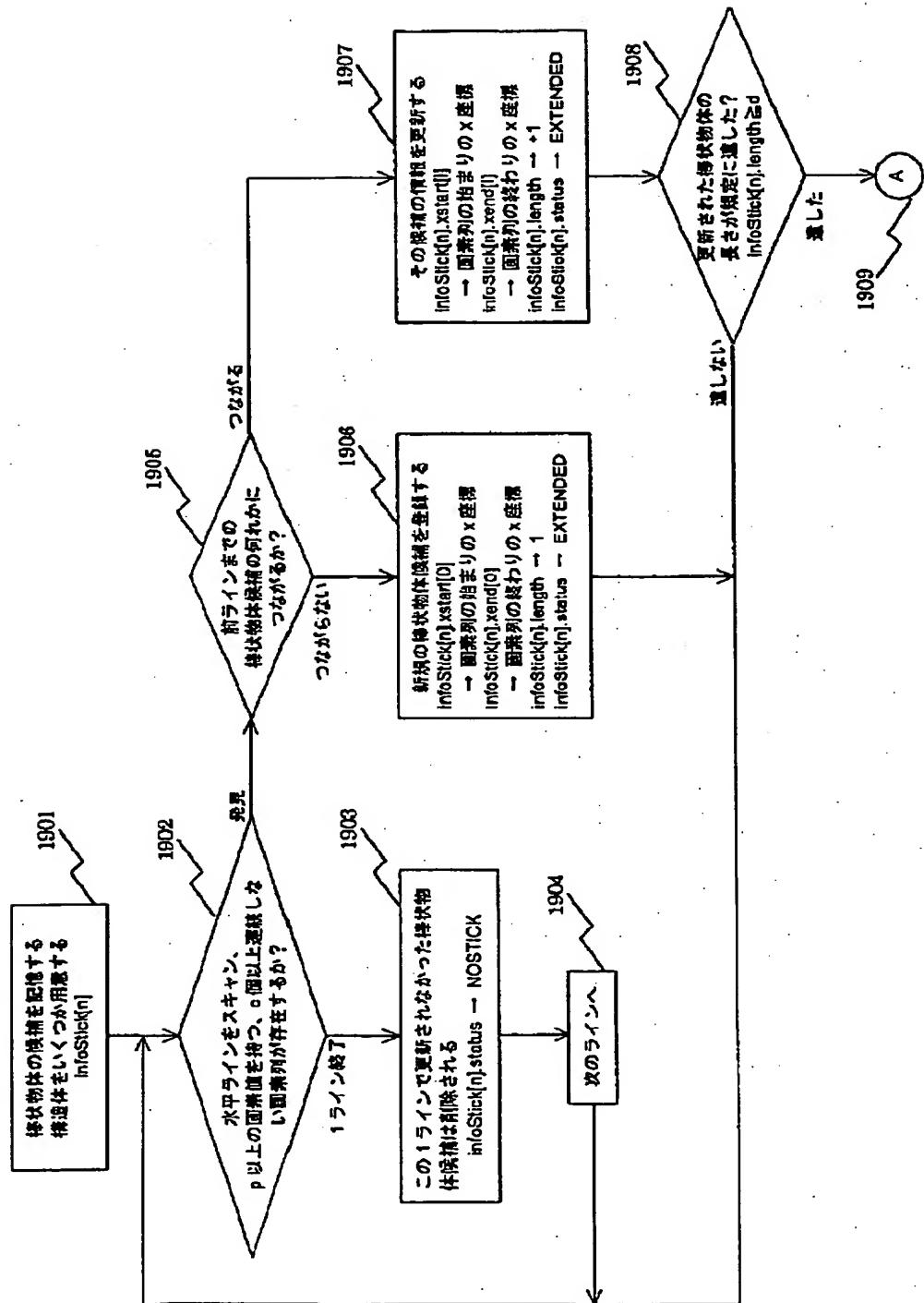
【図15】



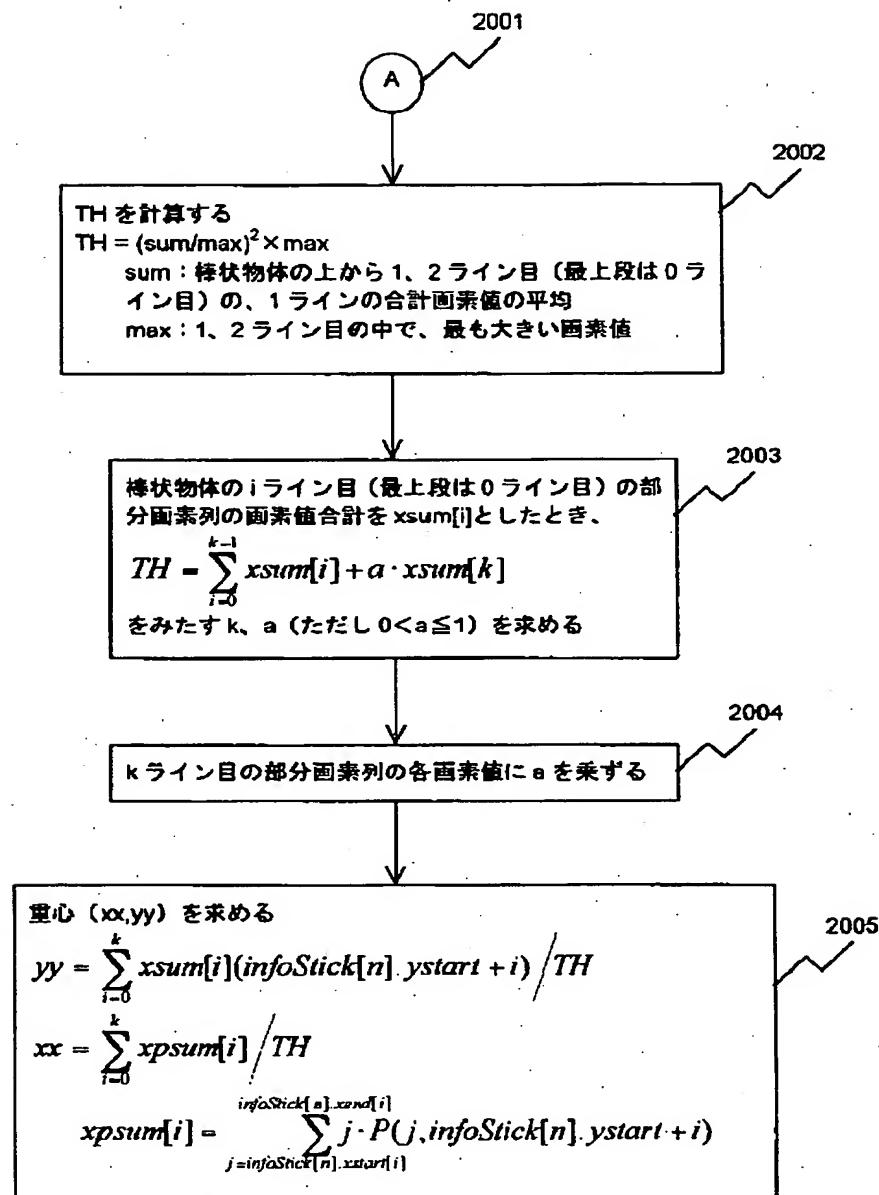
【図18】



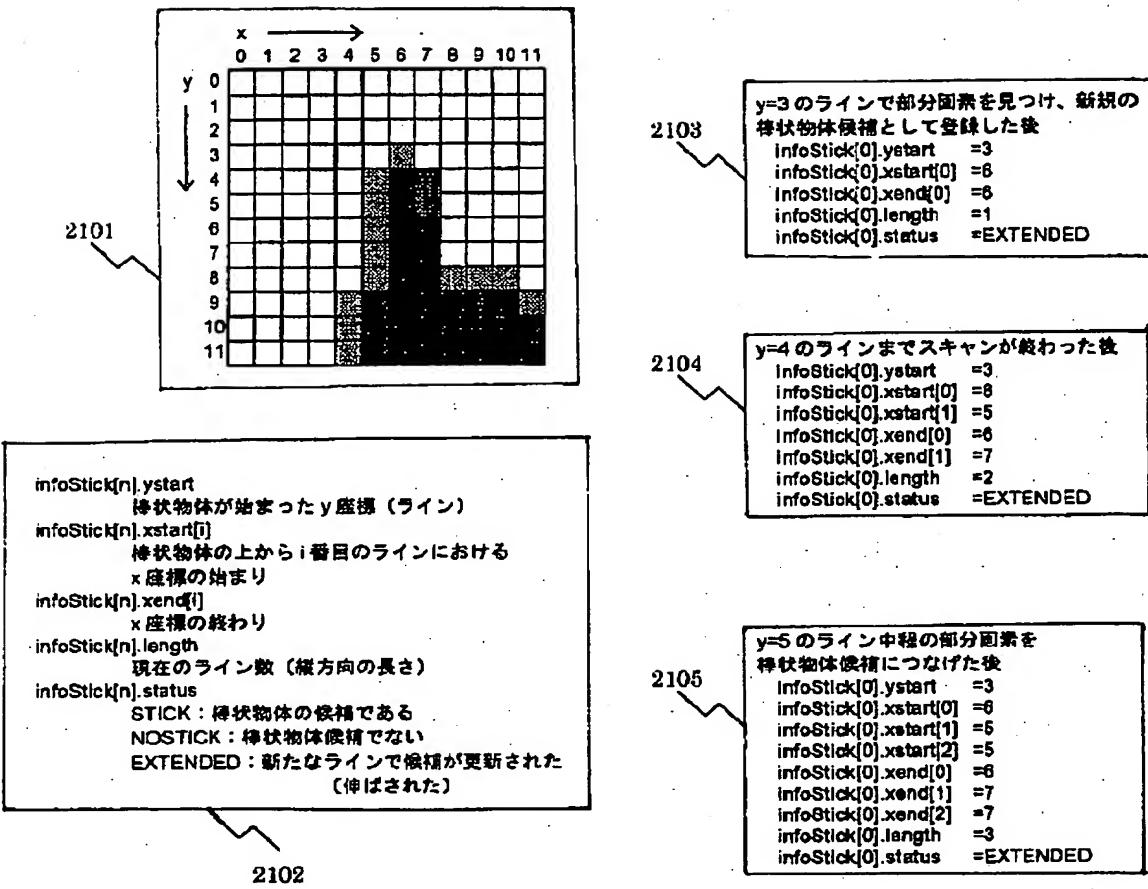
【図19】



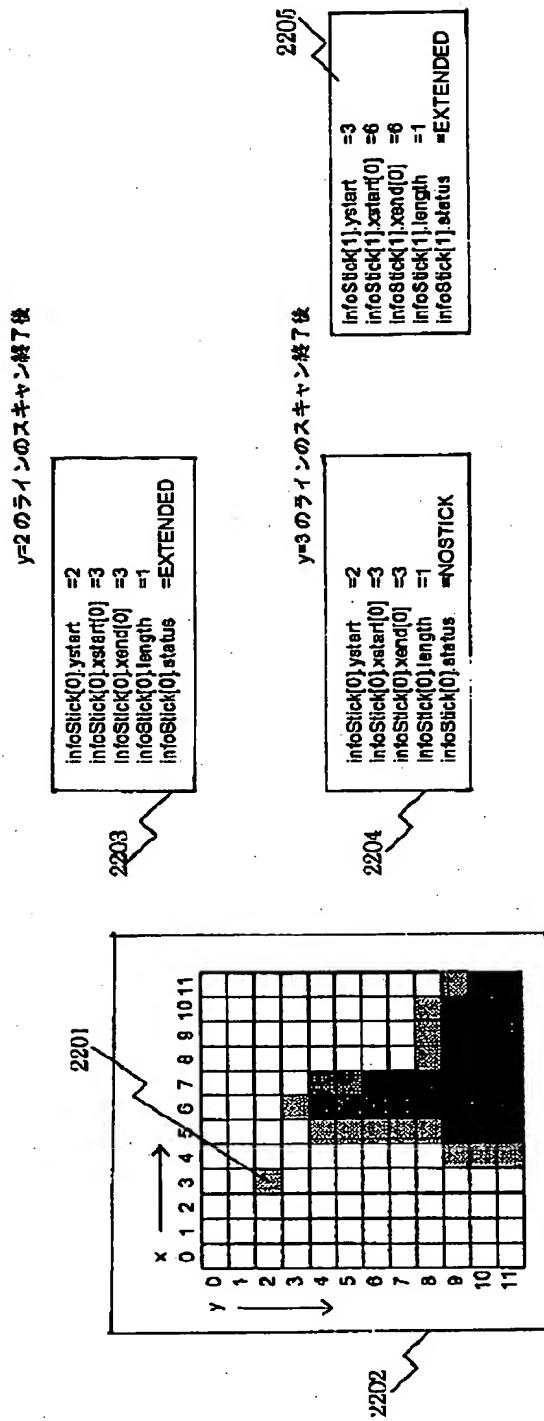
【図20】



【図21】



【図22】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.